Create a Bash Script to Remove a Let's Encrypt Certificate

Let's Encrypt is one of the most popular SSL certificate services, used to get free SSL certificates and automatically manage their renewal. However, there may be situations when you need to remove a Let's Encrypt SSL certificate from a domain. In this article, we will see how to create a Bash script that accepts a domain name as a parameter, checks if a Let's Encrypt certificate exists for that domain, and if so, removes it.

Our goal is to create a script that:

1. Accepts a domain as a parameter.
2. Checks if a Let's Encrypt certificate exists for that domain.
3. If it does, removes it.

```bash
#!/bin/bash

# Verify that the user is running the script as root
if [ "$EUID" -ne 0 ]; then
  echo "Please run the script as root or using sudo."
  exit 1
fi

# Check if a domain was passed as an argument
if [ -z "$1" ]; then
  echo "Error: No domain specified."
  echo "Usage: $0 domain.com"
  exit 1
fi
```

```bash
# Assign the domain to the variable
DOMAIN=$1

# Check if a certificate exists for the domain
if [ -d "/etc/letsencrypt/live/$DOMAIN" ]; then
  echo "Certificate found for domain $DOMAIN."

  # Revoke the certificate
  certbot revoke --cert-path
/etc/letsencrypt/live/$DOMAIN/cert.pem --reason
superseded

  # Check if revocation was successful
  if [ $? -eq 0 ]; then
    echo "Certificate successfully revoked."

    # Remove files associated with certificate
    certbot delete --cert-name $DOMAIN

    if [ $? -eq 0 ]; then
      echo "Successfully removed certificate and
associated files for $DOMAIN."
    else
      echo "Error removing certificate files for
$DOMAIN."
    fi
  else
    echo "Error revoking certificate for $DOMAIN."
  fi
else
  echo "No certificate found for domain $DOMAIN."
fi
```

Script explanation:

1. **Checking root user**: The script starts by checking if the user running it has root privileges. This is necessary because SSL certificate management requires elevated permissions.
2. **Domain check as parameter**: If the user does not specify a domain, the script exits with an error and shows how to use it correctly.
3. **Certificate existence check**: The script checks if a directory for the specified domain exists in `/etc/letsencrypt/live/`, which is where Certbot stores active certificates.
4. **Certificate revocation**: If the certificate exists, the script uses `certbot revoke` to revoke the certificate. This command requires the path to the certificate, which is specified with the `--cert-path` option.
5. **Deleting associated files**: After revocation, the script uses the `certbot delete` command to remove the certificate files from the server.

Example usage:

```
sudo ./remove_cert.sh domain.com
```

Conclusion

This Bash script provides a simple way to check for the existence of a Let's Encrypt certificate for a domain, and if present, revoke it and remove its associated files. This can be useful when removing domains, replacing certificates, or simply keeping the server clean. Always remember to test and back up before performing this type of operation in production environments.